AD-A031 167 FEDERAL AVIATION ADMINISTRATION WASHINGTON D C SYSTE--ETC F/G 17/7
CENTRAL FLOW CONTROL COMPUTER PROGRAM SPECIFICATIONS. VOLUME V.--ETC(U)
SEP 76 R BALES, G BEEKER, C BROGLIO, F CASEY

UNCLASSIFIED FAA-RD-76-157-VOL-5 NL

1 OF 1
AD
A031167

END
DATE
FILMED
11-76

Report No: FAA-RD-76-157, V

CENTRAL FLOW CONTROL COMPUTER PROGRAM SPECIFICATIONS: VOLUME V EXECUTIVE SUBSYSTEM SPECIFICATION

Central Flow Control Design Team
Federal Aviation Administration

September 1976
Final Report

Document is available to the public through the National Technical Information Service, Springfield, Virginia 22161.

Prepared for

U.S. DEPARTMENT OF TRANSPORTATION
FEDERAL AVIATION ADMINISTRATION
Systems Research & Development Service
Washington, D.C. 20590

NOTICE

This document is disseminated under the sponsorship of the Department of Transportation in the interest of information exchange. The United States Government assumes no liability for its contents or use thereof.

Technical Report Documentation Page

1. Report No. (14) FAA-RD-76-157[redacted]—Vol-5	2. Government Accession No.	3. Recipient's Catalog No. (11)
4. Title and Subtitle (6) Central Flow Control Computer Program Specifications. Volume V. Executive Subsystem Specification.		5. Report Date Sept[redacted] 1976
		6. Performing Organization Code ARD-102
7. Author(s) Central Flow Control Design Team		8. Performing Organization Report No. 111-102
9. Performing Organization Name and Address U.S. Department of Transportation, Federal Aviation Administration, Systems Research and Development Service, Washington, D.C. 20590		10. Work Unit No. (TRAIS)
		11. Contract or Grant No. 111-102
12. Sponsoring Agency Name and Address U.S. Department of Transportation, Federal Aviation Administration, Systems Research and Development Service, Washington, D.C. 20590		13. Type of Report and Period Covered (9) Final Report,
		14. Sponsoring Agency Code ARD-102

15. Supplementary Notes (10) R. Bales, G. Beeker, C. Broglio, F. Casey Y. Chao   P V/vi   (12) 47p.

16. Abstract

This report contains the specifications for the executive subsystems of the Central Flow Control Computer Program. This report provides a detailed functional specification of the system's resource management, error analysis, reconfiguration, startup and startover, system analysis recording, input/output message, testing, and supervisor call components.

17. Key Words Operating System, Control Program, Executive, Monitor, Functional Specification, Flow Control, En Route Monitor	18. Distribution Statement Document is available to the public through the National Technical Information Service, Springfield, Virginia 22161

19. Security Classif. (of this report)	20. Security Classif. (of this page)	21. No. of Pages	22. Price
Unclassified	Unclassified	49	

Form DOT F 1700.7 (8-72)   Reproduction of completed page authorized

340170
6B

METRIC CONVERSION FACTORS

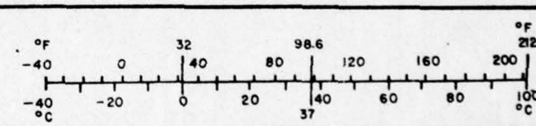
Approximate Conversions to Metric Measures

Symbol	When You Know	Multiply by	To Find	Symbol
		LENGTH		
in	inches	*2.5	centimeters	cm
ft	feet	30	centimeters	cm
yd	yards	0.9	meters	m
mi	miles	1.6	kilometers	km
		AREA		
in^2	square inches	6.5	square centimeters	cm^2
ft^2	square feet	0.09	square meters	m^2
yd^2	square yards	0.8	square meters	m^2
mi^2	square miles	2.6	square kilometers	km^2
	acres	0.4	hectares	ha
		MASS (weight)		
oz	ounces	28	grams	g
lb	pounds	0.45	kilograms	kg
	short tons (2000 lb)	0.9	tonnes	t
		VOLUME		
tsp	teaspoons	5	milliliters	ml
Tbsp	tablespoons	15	milliliters	ml
fl oz	fluid ounces	30	milliliters	ml
c	cups	0.24	liters	l
pt	pints	0.47	liters	l
qt	quarts	0.95	liters	l
gal	gallons	3.8	liters	l
ft^3	cubic feet	0.03	cubic meters	m^3
yd^3	cubic yards	0.76	cubic meters	m^3
		TEMPERATURE (exact)		
°F	Fahrenheit temperature	5/9 (after subtracting 32)	Celsius temperature	°C

*1 in = 2.54 (exactly). For other exact conversions and more detailed tables, see NBS Misc. Publ. 286, Units of Weights and Measures, Price $2.25, SD Catalog No. C13.10:286.

Approximate Conversions from Metric Measures

Symbol	When You Know	Multiply by	To Find	Symbol
		LENGTH		
mm	millimeters	0.04	inches	in
cm	centimeters	0.4	inches	in
m	meters	3.3	feet	ft
m	meters	1.1	yards	yd
km	kilometers	0.6	miles	mi
		AREA		
cm^2	square centimeters	0.16	square inches	in^2
m^2	square meters	1.2	square yards	yd^2
km^2	square kilometers	0.4	square miles	mi^2
ha	hectares (10,000 m^2)	2.5	acres	
		MASS (weight)		
g	grams	0.035	ounces	oz
kg	kilograms	2.2	pounds	lb
t	tonnes (1000 kg)	1.1	short tons	
		VOLUME		
ml	milliliters	0.03	fluid ounces	fl oz
l	liters	2.1	pints	pt
l	liters	1.06	quarts	qt
l	liters	0.26	gallons	gal
m^3	cubic meters	35	cubic feet	ft^3
m^3	cubic meters	1.3	cubic yards	yd^3
		TEMPERATURE (exact)		
°C	Celsius temperature	9/5 (then add 32)	Fahrenheit temperature	°F



LIST OF MEMBERS OF THE CENTRAL FLOW CONTROL DESIGN TEAM

	R. Bales	MITRE	T. McCann	AAT-370
	G. Beeker	MITRE	M. McCrory	ARD-141
(1)	C. Broglio, Ph.D.	ARD-102	M. Medeiros	TSC-622
	F. Casey	ARD-141	O. Morganstern	MITRE
	Y. Chao, Ph.D.	FEDSIM	J. Noyes	AAF-630
	R. Davis	FEDSIM	W. Reed	AAF-643
	G. Ellison	ARD-142	J. Richardson	AAT-370
	H. Gabrielli	MITRE	A. Robb	TSC-622
	T. Guyc	BDM	D. Rozzano	ARD-142
	V. Hallows	AAT-510	A. Severino	ARD-142
(2)	T. Hannan	ARD-102	J. Siedsma	AAT-510
	L. Jarze	FEDSIM	R. Watson	MITRE
	G. Kershaw	BDM	H. Whang	TSC-622
	P. MacDonald	TSC-622	G. Wright	FEDSIM
	V. Maglione	TSC-622	R. Wright	TSC-622

(1) Chief Programmer
(2) Back-up Programmer

ACKNOWLEDGEMENT

This specification series was prepared by the members whose names are listed above. In performing this work these members were supported by their respective organizations. Inclusion in the above list, however, does not necessarily imply that either the individual or his organization endorses the report.

CONTRIBUTING ORGANIZATIONS

AAF - Airway Facilities Service, Federal Aviation Administration
AAT - Air Traffic Service, Federal Aviation Administration
ARD - Systems Research and Development Service, Federal Aviation Administration
BDM - The BDM Corporation, Vienna, Virginia
FEDSIM - Federal Computer Performance Evaluation and Simulation Center
MITRE - The MITRE Corporation, McLean, Virginia
TSC - Transportation Systems Center, Department of Transportation

CHANGE HISTORY

This CPFS is the original edition. Revision Levels, change levels, and applicable dates are indicated below

Revision Level	Change Level	Date	Change/Title Baseline

F. Casey, ARD-141
W. Reed, AAF-643
N. Ingham, AAT-542 (Principal Consultant)
M. Kinzer, ARD-141, (Text Edit Operator)
06/19/76

TABLE OF CONTENTS

1.0 INTRODUCTION

This document specifies the functions of the NAS CFC Executive program.

1.1 BACKGROUND

The NAS EnRoute A3d2.2 Monitor program will be the development base for the CFC Executive program. This specification therefore, was prepared using NAS-MD-317, the EnRoute Monitor specification, as a base. It specifies those functions in the EnRoute monitor program which are to be retained and those which are to be added, deleted or modified in order to convert the Monitor for use as the CFC Executive program.

1.1.1 Scope

The functional scope of the CFC Executive is greater than that of the EnRoute Monitor and this specification reflects that fact. For example the CFC Executive is assigned additional functions related to the processing of Inputs and Outputs. In the EnRoute system these functions are accomplished by application programs. The CFC Executive is also assigned functional responsibility for all common system subroutine functions; in the EnRoute System these responsibilities are divided between the applications and Monitor subsystems.

This specification also defines every Input and Output message and every Supervisor Call which is the functional responsibility of the Executive. The Executive data base is described and specified. In addition documentation associated with this specification is described and specified.

1.1.2 Methodology

A comparison of functional changes is facilitated by retaining the general outline and indexing of NAS-MD-317. In addition, the modifications to the outline and index will permit an easy transition to a stand-alone document when frequent references to NAS-MD-317 are no longer required.

The addition of functions is readily apparent from both the index and the text. Deletions are also apparent either by their omission or by the note "N/A' in the index and text.

Major changes within retained functions however, are noted as ("Revision Note # n" with the parentheses closed at the end of the text describing the change. There are Revision Notes in the following subsections:

1. 2.1.3

2. 2.2.3.2

3. 2.3

4. 2.3.2x

5. 2.4.4.4

6. 3.2.1.3

Subsections of several EnRoute A3d2.2 documents are incorporated by reference in this specification. The actual text in these documents is not duplicated in this specification. These references, nevertheless, have the same force as the original text in this specification.

1.2 SYSTEM CONTROL FUNCTIONS

System Control, which is the function of the Executive program, interfaces the NAS CFC equipment with the NAS CFC Applications and Data Base Management functions. The major subfunctions of System Control are:

a. Resource Management

   This subfunction manages computation capacity, storage, and I/O.

b. Error Analysis and Program Controlled Reconfiguration

   This subfunction provides for error analysis and rerouting in the case of I/O errors, and error analysis and reconfiguration in the case of element errors.

c. Startup/Startover

   This subfunction provides for system initialization at IPL time and for two modes of recovery from element errors; Resume Mode when the data base is unaffected by the error, and Re-establish Mode when the data base is affected. Recovery recording preserves the data base for startovers, by periodically transferring the contents of critical system tables to "safe storage".

d. System Analysis Recording (SAR)

   This subfunction provides the capability to record data concerning significant system events, on a designated magnetic tape.

e. Executive Input/Output Messages

This subfunction accommodates a series of messages relating to Executive functions such as control of I/O, element configuration and error processing.

f. Test and Debugging Aids

This subfunction provides capabilities which are necessary or helpful for system development and installation. Some examples of the required capabilities are simulated inputs, program abort dumps and program modification facilities.

1.3 DESIGN APPROACH

The fundamental architecture of the EnRoute Monitor will be retained in the code which is modified for use as the CFC Executive. Some additions may be accomplished by the addition of modular code to clear stubs in the EnRoute Monitor. Some deletions may be accomplished by the relatively simple deletion of integral code modules within the EnRoute Monitor. Other additions and deletions and changes will require the modification of code within any of the levels of identifiable code modules within the Monitor. In any event, all new design and code will conform to the conventions specified in the Request for Proposal.

This specification is conservative with regard to the retention of EnRoute Monitor functions, messages and SVC calls. Additional candidate functions for deletion or modification may therefore become apparent during ensuing phases of this effort.

2.0 RESOURCE MANAGEMENT

The purpose of this section is to specify general requirements for efficient use of system resources; computing capacity, storage, and I/O channels and devices. This section also specifies the requirements for management of time-related functions, including maintenance of system times and calendars.

In this section, the term "Computing Resource" refers to both the Computing Element (CE) and the Input/Output Control Element Processor (IOCE-P) unless otherwise noted.

2.1 SCHEDULING

The scheduling function is based on the concepts of Interrupt Processing, Multiprogramming and Multiprocessing.

2.1.1 Interrupt Processing

The Executive gains control of Computing Resources through interrupts. Interrupts identify events that may affect resource utilization. The concept of interrupt processing is fundamental to Executive operation.

The Executive scheduling function gives control of Computing Resources to programs requiring execution. A Computing Resource executes the scheduled program until an interrupt occurs. Gaining control from the interrupt, the executive saves the environment of the interrupted program and analyzes the cause of the interrupt. After taking appropriate action (e.g., noting channel availability after a completed I/O operation) control will be transferred to any higher priority programs eligible for execution before it is returned to the interrupted program. This is the normal exchange of control between the Executive program and scheduled programs.

2.1.2 Multiprogramming

Multiprogramming is a method of redistributing computing resources during periods when programs are suspended awaiting system resources before resuming execution.

Through integration with interrupt processing, multiprogramming provides the capability to transfer control of a Computing Resource to a program other than the one interrupted.

2.1.3 Multiprocessing

Multiprocessing is a method of dynamically distributing the processing work load among the Computing Resources in the operational system. Distribution is accomplished by treating each operational Computing Resource as a resource which can be allocated to any of the processing functions.

(Revision Note #1

Allocation of Computing Resources is dynamic for both Computing Elements and Input/Output Control Element Processors. Each IOCE-P shall be capable of executing, in a multiprogramming mode, a specific set of schedulable subprograms which are resident in the Maintenance and Channel (MACH) storage of the same IOCE. The IOCE-P's shall also be capable of processing an eligible subset of SE-resident subprograms in a multiprocessing mode. The program shall be capable of operating with any combination which includes from one (1) to four (4) Compute Elements and two (2) Input-Output Control Elements.

Design shall be modified to permit one or more subprograms of a selected subset of schedulable subprograms to be simultaneously executed by more than one Computing Resource. The design and implementation of all schedulable subprograms shall allow for the

possibility of any of these subprograms being incorporated in this subset.)

2.2 STORAGE MANAGEMENT

A multiprocessing, multiprogramming environment requires central control of common storage resources in order to insure adequate protection of storage areas and to provide for the management of these resources. Storage resources are requested via specialized Executive services.

If a requested storage resource is available, the Executive facilities will allocate the resource and return control of the computing element to the requester. When a requested storage element is unavailable, the operation of the requesting subprogram is suspended until the resource becomes available. When a requested storage resource does not exist, the requester is immediately terminated and purged of any resources already obtained. When a program no longer requires an acquired storage resource, it indicates to the Executive that the resource is again available by returning control of the executing Computing Resource to the Executive.

The following three (3) types of storage resources are required:

2.2.1 Serially Reusable Storage Resources

A serially reusable storage resource is an area of storage that contains data or code used by two (2) or more subprograms and that cannot be accessed simultaneously without possible damage to the system data base. The Executive provides requesting subprograms with exclusive access (for modification) or controlled shared access (for reference) to such storage areas.

2.2.2 Working and Communication Storage

Blocks of main storage will be allocated by Executive services. These blocks may be used in the performance of any functions which require storage not internally allocated to subprograms.

2.2.3 Program and Input/Output Device Queues

2.2.3.1 Normal Queuing

A queue is used to identify communication paths to other subprograms and to input/output devices. For any particular subprogram or input/output device, a limited number of queues are available for external communication. Data to be communicated reside in temporary working storage obtained by the use of Executive storage allocation facilities. A subprogram transfers control of an allocated block of storage to another subprogram or device by requesting an appropriate queue entry under Executive

control. Facilities for allocating and deallocating such queues will be provided.

(Revision Note #2

2.2.3.2 Auxiliary Queuing

The Executive program will provide the capability to logically and physically extend selected queues from core storage to auxiliary storage (disk). The queues to be so extended will be selected during the design and implementation phases.)

2.2.4 Storage Management Conventions

Hierarchical requirements for resource allocation and deallocation will be enforced by the Executive to decrease the possibility of mutual suspension of subprograms.

2.2.5 Buffering of Programs and Data

The CFC Executive will support the buffering of program and data modules from disk to main storage in a way that allows the same unit of main storage to be used for different modules at different times depending on the requirements of the applications subsystem.

2.3 INPUT/OUTPUT CONTROL

Input/Output (I/O) control is centralized in a set of Executive routines. Executive I/O control has three main functions:

a. To handle I/O requests, which are requests for the execution of channel programs.

b. To handle I/O interrupts, which result from the execution of channel programs, from operator intervention and from certain error conditions.

((Revision Note #3

c. To present inputs to and accept outputs from the applications and data base subsystems in a common internal format.)

The Executive is capable of controlling a variety of types of I/O operations and equipment, each having different data and timing characteristics. The Executive I/O routines initiate all I/O operations and record data about I/O channel and device utilization. I/O completion interrupts and equipment fault indications are handled by this function. The I/O control function processes all I/O requests and maintains status

information about the request from the time the request is accepted until it is completed.

2.3.1 Input/Output Requests

All requests for input/output operations from the applications subsystem are processed by the Executive. An I/O request is represented by a block of storage which is leased from the general storage pool. Requests are chained in each device queue. I/O requests originate from SVCs issued by subprograms and from certain Executive routines that service the individual requirements of specific devices.

A number of I/O operations may take place simultaneously in the operational system because of the use of multiple subchannels. The Executive monitors the status of each I/O device and channel independently and executes pending I/O requests as soon as each device and channel is available.

Since system I/O devices transmit or receive data at a slower rate than the internal processing speed of the computer, buffer space is provided, either in leased blocks from the general storage pool or in other areas of core storage.

The Executive reads input data into the CCC either automatically or in response to subprogram requests. Input data are collected in one of the buffer areas as they are received until a complete message is transmitted. Output operations are requested by subprograms when necessary. The subprogram generates the output data, stores the data in an appropriate output buffer, and transmits an output request to the Executive by issuing an SVC instruction. The Executive may also generate its own output request. When an I/O operation is requested and the device or channel is busy, the request is saved and then is carried out when the device or channel becomes available. Subprogram operation may be optionally suspended during a requested I/O operation.

2.3.2 Input/Output Interrupt Processing

All I/O interrupt processing is done by the Executive. When an I/O interrupt is received, the Executive determines which of the various subchannels generated the interrupt and executes an appropriate servicing routine for the particular device type.

When an input operation is successfully completed, the Executive interrupt processors set the appropriate indicators to allow scheduled subprograms to continue processing of the message. For an output operation, the Executive interrupt processors set the status of the I/O operation to complete. If the requesting subprogram had been suspended awaiting completion of the output operation its suspended status is cleared. For both input and

output, the Executive then starts any additional I/O operations for the channel or device.

(Revision Note #4

2.3.2x Application I/O Interface

The requirement to present inputs to the applications and data base subsystems in a common system format and accept outputs in the same format dictates that the Executive program perform the following functions:

a. Delete input transmission control data

b. Attach processing control header to inputs

c. Translate input device type data code to the internal processing code (Upper Case EBCDIC).

d. Route inputs based on the indicated applications, data base or executive functions.

e. Accept system output data from the applications, data base or executive functions.

f. Decode the processing control header accompanying the output message.

g. Acquire temporary storage resources for output message routing to multiple destination devices and device types.

h. Translate output message data from the internal processing code to the output device type data code.

i. Attach any required transmission control data to the output message data

j. Queue the output message to the appropriate device.)

2.3.3 I/O Devices Supported

Executive I/O routines control the input and output operations of the following device types:

a. IBM 2400 Series Tape Drives

b. IBM 1403 Printer

c. IBM 1402/2540 Card Punch

d. IBM 1402/2540 Card Reader

e. IBM 1052 Printer-Keyboard (IOT)

f. N/A

g. N/A

h. N/A

i. Output Only Teletypewriter

j. Interfacility Input (INTI) and Interfacility Output (INTO) Equipment

k. N/A

l. N/A

m. Coded Time Source (CTS)

n. N/A

o. N/A

p. IBM 2314 Direct Access Storage Facility (Disk)

q. N/A

r. N/A

s. Medium Speed Printer

t. Display Terminal Equipment

2.3.4 Logical Addressing

A logical device may be referred to by either an alphanumeric or numeric identifier. The alphanumeric identifier is a mnemonic that describes the logical device function. This identifier must begin with a letter and must be 2-6 alphanumeric characters. The numeric identifier, although available to the operations and maintenance personnel, is used mainly within the program to define the logical device-physical device relationship. There is one-to-one correspondence between alphanumeric and numeric identifiers.

A physical device may be referenced by either its hexadecimal equipment address (SIO level) or by an alphanumeric identifier.

This identifier must begin with a letter and must contain 2-6 alphanumeric characters.

The Executive provides the capability to change logical device-physical device relationships dynamically.

2.4 TIME-RELATED CONTROL AND SERVICES

The Executive initializes and maintains system clocks and calendars. In addition it provides the following; time dependent scheduling, timeout services for subprograms and I/O devices and a time-tagged system trace.

2.4.1 Time Dependent Scheduling

The Executive periodically initiates those subprograms that require scheduling based on elapsed time. The Executive schedules subprograms according to preset parameters or according to time parameters that are subprogram generated and transmitted to the Executive in an SVC request.

2.4.2 Timing Analysis Records (TAR's)

Timing Analysis Records (TAR's) which are generated at major branch points in the Executive program, provide trace and timing information on the occurrences of significant system events such as the operation of subprograms and I/O devices.

2.4.3 Subprogram and /O Timeouts

2.4.3.1 Subprogram Timeouts

The Executive calculates the elapsed execution time for each instance of a subprogram's execution. Subprogram execution time is kept in units of 1/300 of a second, to an accuracy of 1/60 of a second. This time is optionally compared by the Executive against a predefined maximum allowable execution time for the particular subprogram. If the subprogram overruns its defined maximum execution time, that instance of execution of the subprogram is aborted. This function is enabled or disabled via Executive adaptation.

2.4.3.2 I/O Channel and Device Timeouts

The Executive I/O routines maintain timing information for I/O channel and device utilization. The Executive calculates a maximum allowable time for each I/O operation, to ensure that no I/O device or channel is lost to the system because of an I/O completion failure. (Input operations for INTI are exceptions because of the constant read-select requirement). If a device

exceeds its allowable I/O execution time, the Executive terminates the operation and calls I/O error analysis routines.

All 1052 input operations may be interrupted by the Executive in order to give output operations priority. After all waiting outputs for the particular 1052 are completed, the 1052 will be read-enabled to permit continuation of the interrupted input.

2.4.4 System Clocks and Calendars

2.4.4.1 Coded Time Source (CTS)

The source for real-time is a set of two duplexed Coded Time Sources (CTSs) with auxiliary equipment. Each CTS provides binary coded (BCD) time in hours, minutes and seconds over a multiplexor channel interface. A CTS is used by the Executive to initialize internal clocks at startup time. The monitor selects one CTS to provide real time over auxiliary interfaces to other external equipment.

Both CTS's are read by the Executive even though only one is used as the primary time source. This is necessary to detect a failure in either CTS or a time drift between the two CTS's. CTS monitoring is done by a continuously executing channel program.

A CTS is considered out of service if one or both of the following conditions occur:

a. Time cannot be successfully requested from the CTS.

b. Input CTS time contains error flags.

2.4.4.2 Internal Clocks

Internal clock time is updated by the Executive, using the interval timer in a designated Computing Element. This interval timer is set and reset to cause a timer interrupt every half second. Internal time is synchronized to a CTS, to within plus or minus one (1) second accuracy. Corrections to internal time are made in small increments to avoid impacting subprogram scheduling. Internal time is never stepped backwards.

2.4.4.3 System Calendars

When system time passes midnight, the system date is updated, including day/month/year, with provision for leap years.

(Revision Note #5

2.4.4.4 Epoch Time

The Executive program will maintain Epoch Time as an integer value for use in internal processing. This value will represent current year, month, day, hour, minute and second.)

3.0 ERROR ANALYSIS AND PROGRAM CONTROLLED RECONFIGURATION

Error analysis for NAS CFC encompasses both I/O error analysis, concerning malfunctions which are associated with I/O operations and element error analysis, concerning malfunctions which occur within the reconfigurable CCC elements. I/O error analysis provides for detection and reporting of I/O errors and operational recovery from I/O device failures. Element error analysis provides for identification and reporting of element errors and programmed recovery from element failures.

Program controlled reconfiguration encompasses both program generated reconfiguration due to element failures and manually requested reconfiguration of the CCC.

3.1 I/O ERROR ANALYSIS

The CCC and all attached peripheral equipment contain error detection circuitry to insure that malfunctions in the I/O subsystem are detected at the earliest possible stage. Whenever the hardware detects a malfunction within the I/O subsystem, this fact is communicated to the program with appropriate status and sense indications. The CFC Executive insures that the following major tasks are performed in response to an I/O Error:

a. Analyze the error environment provided by the hardware

b. Identify the failing device, control unit, adapter or channel

c. Determine if the failure is transient or solid

d. Provide information to maintenance and operations personnel, in hardcopy form and on tape, describing the error environment

e. Initiate retry and rerouting procedures

3.1.1 Detection of I/O Errors

Errors within the I/O subsystem are detected in four ways:

a. By a device

b. By a control unit or an adapter

c. By an I/O channel

d. By the program

Indications of specific device error conditions are provided by a device, through sense lines to its control unit or adapter. A sense operation is performed by the program to obtain the detailed sense data provided by the device. Sense data are device dependent and indicate either equipment failures or manual intervention conditions such as not ready, out of forms, paper jam, etc.

Malfunctions of the common logic within PAM's, TCU's, DCU,s and other control units set flags in sense registers of the control unit. These flags are available to the program through a sense operation. They define errors which include; lack of response by a device (timeout), parity checks and command reject. Furthermore, in the case of PAM's, TCU's and DCU's, out-of-tolerance (OTC) conditions are indicated by sense bits. The OTC condition indicates that the internal temperature of the element has exceeded an upper bound. This error is handled as an element error as described in paragraph 3.2.

I/O errors associated with equipment malfunctions detected by a channel, set one of the following CSW bits; channel control check bit, channel data check bit or interface control check bit. In the case of the channel check bit and the interface control check bit, the associated selector channel is logged out. These check bits generally indicate a malfunction in either the IOCE or in the interface between the IOCE and the control unit. When any of the three check bits are set the error is handled as an element error as described in paragraph 3.2.

I/O operations are monitored by the program to insure completion of the operation within an appropriate time period. Immediately prior to the start of an I/O operation, the program calculates an estimate of the time required to complete the operation and then starts a timeout. Should the timeout expire before an I/O completion interrupt, the device is considered to have failed and retry procedures are initiated if applicable.

3.1.2 I/O Check Reports

The I/O Check Report is used to notify operations and maintenance personnel that a malfunction is detected in the I/O equipment subsystem by either the equipment or the program. Three types of I/O Check Reports are produced by the Executive:

a. Initial I/O Check Report

b. Summarized I/O Check Report

c. Condensed I/O Check Report

When the CFC Executive detects an I/O equipment failure, all data describing the failure are saved in storage for later use in producing an I/O Check Report. Each retry of the operation is counted and, upon final resolution of the operation, the I/O Check Report is put out. The following information is included in the I/O Check Report. (See Section 6 for details of format and routing.):

a. Date and time of error

b. Failing device, control unit or channel

c. Detection method, i.e., I/O error interrupt, program timeout, condition code

d. Error environment, e.g., channel address word (CAW), sense data (if present) and selector channel logout (if present)

e. Number of retries

f. Indication of whether the operation is successful or abandoned

The Initial I/O Check Report is put out immediately upon final resolution of either a single error on a device or the first of a series of errors on a device, for a given I/O request. This report contains all the information specified in the preceding six items.

A Summarized I/O Check Report is put out for a series of identical errors received from the same device, control unit or adapter over a parameter period of seconds. A value of zero for the parameter is used by the program to eliminate Summarized I/O Check Reports and to put out an Initial I/O Check Report for each detected error. A separate time parameter is stored for each device type in the I/O subsystem (i.e., one parameter for all TTYS, another parameter for all IOTs etc.). Each parameter is initially set by adaptation and may be modified dynamically by an input message. (See Section 6)

A Summarized I/O Check Report contains the same data as an Initial I/O Check Report and is put out only if subsequent errors are detected within the I/O Check Report Summarization Interval and are identical to the errors described in a preceding Initial I/O Check Report. If an error is encountered which has not been described in a previous Initial I/O Check Report, it will result in an Initial I/O Check Report even if other errors from the same device are currently being summarized. The Summarized I/O Check Report is put out upon expiration of the summarization interval for that device type. The program considers any subsequent I/O

malfunctions to be initial errors. All of the following criteria are used in determining if errors are identical:

a. The device address is the same

b. The command from the CCW is the same

c. The status from the CSW is the same

d. The sense data are the same

The above criteria are used for determining if errors are identical during retry operations and are also used for summarization purposes.

A Condensed I/O Check Report is put out at the same time as either an Initial I/O Check Report or a Summarized I/O Check Report. It provides only the most significant information concerning the I/O malfunction.

3.1.3 Recovery from I/O Failures

Upon detection of an I/O malfunction the Executive saves all information about the malfunction. A parameterized retry procedure is initiated. Back up channels are tried if they are available. If back up channels are unavailable or fail, the operation is rerouted to the available back up device.

3.1.3.1 Device Failures

Device failures are classified as either transient or solid by means of a retry procedure for the I/O operation that resulted in the error. If the retry of the operation proves successful, the error is defined as transient, and an I/O Check Report is put out indicating a successful operation. An I/O failure is assumed transient until n (a predefined value) retries have been unsuccessful on both the primary and backup channels for the device. The number of retries performed is a function of the device type.

Once an I/O device is determined to have experienced a solid failure and an I/O Check Report is issued, the retry procedure is not subsequently utilized on this device until at least one I/O operation has been successfully completed. This means that any messages destined for a failed device are still routed to that device and if the subsequent I/O operations are unsuccessful, rerouting to the backup device is continued witout retry. This procedure provides automatic status monitoring of failed devices.

Certain failures are immediately classified as solid. These are failures that result from the following types of conditions; a device runs out of paper, a paper jam occurs, a file protected

tape is mounted for outputs, etc. These failures always require manual intervention by operations or maintenance personnel (i.e., reload paper, clear paper jam, mount scratch tape, etc.). The program does not attempt any retry of the operation. It immediately reroutes the message. An INT REQ message is put out to identify the device requiring attention. Subsequent messages destined for the same device are tried once to determine if the failure condition is cleared. If the condition is not cleared rerouting takes place without any additional notification. After the condition is cleared the program puts out an INT CLRD message.

3.1.3.2 Channel, PAM, and Control Unit Failures

Recovery from failures of a channel, PAM, or control unit require the rerouting of the operation to a path that does not include the failing circuitry. The I/O subsystem treats channel failures as solid failures and therefore no retry procedures are instituted. An I/O Check Report and a channel logout, if it is available, are put out. This information is also passed to the Element Error Analysis subsystem for further analysis (see Subsection 3.2). The I/O operation is rerouted to a backup channel or a backup device.

Failures of PAMs, TCUs, DCUs, and other control units are indicated by bits set in the sense register within the unit. The program performs a sense operation to retrieve the sense data. If a PAM, TCU, or DCU sets the OTC bit in its sense register, this information is passed to the Element Error Analysis subsystem for reconfiguration.

3.1.3.3 I/O Reroute

I/O operations are rerouted upon detection of a solid error at the device, control unit, or channel level. The basic tasks of the I/O reroute subfunction are to try the I/O operation over a different path to the same device and, if this new path fails, to direct the data to a backup device. The I/O is directed onto each possible path to each device, and then to a backup device, until either a successful transmission is achieved or all alternate paths and devices are exhausted. The backup devices are specified to the program in adaptation tables. Alternate paths to a device are configuration dependent and must be determined by the program. The specific hierarchy of rerouting is as follows:

a. Primary device, primary channel

b. Primary device, alternate channel

c. Backup device, primary channel

d. Backup device, alternate channel

If a solid failure occurs on every available device and routing path an alert message is put out to request manual reassignment of the logical device. Output messages remain queued to the backup device.

3.2 ELEMENT ERROR ANALYSIS

The CCC equipment elements, i.e., CE, IOCE, SE, PAM, TCU and DCU, have the capability to monitor their own errors. The primary Element Error Analysis function is to analyze either element errors which occur in individual elements or errors which occur in the interface between two elements. In addition, Element Error Analysis monitors and reports the occurrence of all element errors and provides for the recovery of the CCC after each element error interruption.

3.2.1 Element Error Isolation

Element errors are signalled to the CCC in the following five (5) ways:

1. Machine Check Interrupts
2. External Interrupts
3. Program Interrupts
4. I/O Interrupts
5. Condition Code

A machine check interrupt is an indication of an abnormal condition pertaining to a CE, IOCE or SE. For a complete list of causes of machine checks refer to the 9020 Principles of Operation Manual. In general, SE and IOCE malfunctions cause a machine check interrupt request to be presented to the controlling CE and an abnormal-condition signal (pulsed Element Check) to be issued to the other CE's in the system as a type of external interrupt. If the malfunction occurs within a CE, that CE accepts its own machine check interrupt and issues an abnormal condition signal (pulsed Element Check) to other CE's in the system.

Element errors detected through external interrupts require examination of the Diagnose Accessible Register (DAR) to determine the type of element check and the identity of the reporting element.

Element errors detected through program interrupts usually occur because of PSA lockout or SE stopped conditions.

Channel data check, channel control check and interface control check are types of I/O interrupts which cause Element Error Analysis to operate. Error analysis is also performed on certain Test and Monitor Adapter (TAM) errors. These include: check stop, multiple priority, single priority, priority transfer, scan check, and CCR parity. CCR parity, which generates an I/O interrupt from the TCU, also causes Element Error Analysis to operate.

Examination of the Condition Code after execution of instructions such as SCON and SATR may reveal element failures.

3.2.1.1 Solid and Intermittent Element Errors

Element errors are divided into three categories:

1. Solid element errors

2. Intermittent element errors.

3. Indeterminate errors

Element Error Analysis uses the following general rule to determine solid element errors:

When a level (continuous) element check is presented to a CE and the element check cannot be cleared through program procedures, the element which issued the element check is charged with a solid element error.

Element Error Analysis uses the following general rule to determine intermittent element errors:

When a pulsed element check or a level element check that can be cleared through program procedures is presented to a CE, the element which issued the element check is charged with an intermittent element error.

The two preceding general rules apply to the isolation of most element errors. The following however, are exceptions to those rules:

a. All Out-of-Tolerance Condition (OTC) errors are treated as solid errors.

b. All On-Battery-Supply (OBS) errors are treated as solid errors (except multiple OBS conditions which indicate a system power failure).

c. IOCE PSBAR parity errors are treated as intermittent errors.

d. Program Interrupts caused by SE failures are treated as solid element failures.

e. When a CE fails to analyze an element error within a parameterized time period, that CE is charged with a solid element error.

f. When there is not a sufficient amount of error information available to charge an element error, then, if possible, the error is charged to an element interface (see Subsection 3.2.1.2). If it is impossible to define and assign an intermittent error, then an indeterminate error is charged.

3.2.1.2 Element Interface Errors

A malfunction which occurs during an attempt by a CE or an IOCE to access an SE may result in an element check flag being set for both the accessing element and the accessed SE. Logouts for both elements are analyzed by Element Error Analysis to determine the source of the malfunction.

If Element Error Analysis determines that only one of the elements involved in an element interface error has experienced a solid failure, that element is charged with a solid error, and no error is charged to the other element. If Element Error Analysis determines that both elements have experienced a solid failure, a solid interface error is charged. For further discussion on analysis of interface errors, refer to Subsection 3.2.2, Error History Analysis.

(Revision Note #6

3.2.1.3 System Power Failures

The Executive program implements the OBS (On-Battery-Supply) capability of the 9020 CCC. This requirement is not mandatory. The associated code therefore, may be deleted or retained optionally, depending on cost vs. benefits considerations to be evaluated during the design and implementation phases.)

3.2.2 Error History Analysis

Since intermittent errors are transient, an error history for each operational element and interface is maintained. It is maintained to detect rapidly recurring intermittent errors, to identify them as solid errors, and to correlate separate interface error analysis results in order to identify the failing element. When the intermittent error count exceeds n (a

predefined value) within t (a predefined time period), that element or interface is declared to have experienced a solid failure.

3.2.3 Element Check Reports

Element Error Analysis monitors and reports the occurrence of all element errors occurring within the operational elements. All data pertinent to each error are saved in storage for subsequent use in composing an Element Check Report. These reports are put out on printers and on tape.

There are two types of Element Check Reports.

1. Detailed Element Check Report

2. Condensed Element Check Report

3.2.3.1 Detailed Element Check Report

Detailed Element Check Reports contain the following information:

a. Date and time of error

b. Failing element(s)

c. Type of error

    1. Intermittent error, single element

    2. Intermittent interface error

    3. Solid error, single element

    4. Solid interface error

    5. Indeterminate error

d. Error counts (when applicable)

e. Type of interrupt, including old PSW (when applicable)

    1. I/O Interrupt

    2. Machine Check Interrupt

    3. External Interrupt

    4. Program Interrupt

Detailed error environment data are available only from those elements configured to the operational subsystem. Logouts contain all data originally specified in the error logging format specification. Each word of logout data is converted to printable binary with the resulting data distributed across one line. The appropriate identification of each field position appears on the preceding print line in tabular form. In addition, the Element Check Report describes the configuration at the time of the error. The detailed contents and format of Element Check Reports are specified in Subsection 6.2.28., NAS-MD-317.

3.2.3.2 Condensed Element Check Report

A Condensed Element Check Report is put out for each Detailed Element Check Report. It provides only the most significant information concerning the element malfunction.

3.2.4 Recovery from Element Failures

Programmed recovery from element failures within the operational system is initiated and executed without manual intervention. Recovery procedures depend on the type of element error experienced (i.e., solid or intermittent) and on the condition of the core storage data following the error.

Solid element errors detected in a failed operational CE, SE, IOCE, PAM, TCU or DCU cause the exchange of a failed operational element with an identical available redundant element. The system has the capability to bootstrap itself from a subnormal complement of elements back to a normal complement.

If Element Error Analysis indicates solid element interface errors and if redundant elements are available for reconfiguration, the program removes both elements involved in the interface from the system and replaces them with the available redundant elements. If only one of the two replacement elements is available for reconfiguration, the program attempts recovery using the one available replacement element.

When there is no available redundant element to replace the failing element, processing is suspended and notification of this is put out on adapted I/O devices. When, however, an operational CE is deleted from the system, processing continues as long as at least one CE remains operational.

Intermittent errors do not result in the reconfiguration of operational elements. They must be processed by Error Analysis and recovery procedures.

In all cases of operational recovery from solid or intermittent element failures, an Element Check Report is put out, on adapted I/O devices, to operations and maintenance personnel.

For each occurrence of an element malfunction, one CE is selected to perform the analysis. The other CE's monitor the selected CE and are used for backup should the selected CE itself experience a malfunction.

If a CE cannot access its preferential storage area (PSA), the CE's preferential storage base address register (PSBAR) is automatically incremented to the address of the alternate PSA area. A bootstrap recovery routine in the SE containing the alternate PSA area provides recovery from primary PSA SE Failures.

After Element Error Analysis has determined the type of element error and has taken appropriate action, the recovery mode must be determined. The modes of recovery are discussed further in Subsection 4.1.

3.3 PROGRAM CONTROLLED RECONFIGURATION OF THE CCC

Program controlled reconfiguration of the CCC is accomplished in response to two types of requests.

a. Program-generated Reconfiguration request

b. Manual Reconfiguration Request Message

Program-generated reconfiguration is initiated by the Element Error Analysis function when it has determined that a solid element error has occurred. This accomplishes the automatic exchange of an available redundant element with the element that has experienced the solid error. If the exchange of elements is successful, an automatic startover is performed. (For a discussion concerning startover refer to paragraph 4.1.2.) If program generated reconfiguration is unsuccessful because of the lack of an available redundant element, processing is suspended. An exception occurs when the failing element is a CE. In this case processing continues if at least one CE is available. In either case, a configuration summary is put out to adapted I/O devices. In all cases of unsuccessful reconfiguration attempts, a notice that insufficient elements are available is put out. Additionally, in the case when processing is continued with a reduced number of CE's, a message indicating degraded mode of operation is put out.

For all the remaining cases of unsuccessful attempts a notice that operational processing has been suspended is put out.

Manual reconfiguration request messages, which are entered on adapted I/O devices, pertain to reconfiguration of both the operational and non-operational systems (Subsection 6.1, NAS-MD-317, provides a complete description of these messages.)

When the exchange of an operational element with an available redundant element is requested and successfully accomplished, notification of this action is put out on adapted I/O devices. When a request is entered to replace an operational element and no redundant element is available the request is rejected. An exception occurs when mandatory replacement of a CE is requested. If there is no available replacement CE but at least one CE will still remain operational, then the requested CE is deleted and processing continues in the degraded mode.

When an operational element is removed from the system it has all its interfaces with other elements terminated both at the removed element and the interfacing element. (The configuration control register (CCR) of the removed element and the CCR's of all interfacing elements are set to provide isolation of the removed element.) All operational CE's retain the capability to set the CCR of the removed element.

There are two program addressable audible alarms that are sounded whenever a reconfiguration of the CCC is attempted. A buzzer alarm is used to alert operations and maintenance personnel of successful response to program-generated or manual reconfiguration requests. A bell alarm alerts operations and maintenance personnel to the suspension of operational processing. The bell alarm is accompanied by a printout on an adapted I/O device. This printout indicates the action required for recovery.

Manual reconfiguration of the non-operational system is accomplished under control of the operational system.

4.0 STARTUP/STARTOVER

The Startup/Startover subfunction accomplishes the following:

a. Initialization of the CFC System at the start of data processing activities

b. Reinitialization of the CFC System following a temporary suspension of processing.

c. System recovery following CCC Element errors

This section specifies the requirements for the Startup/Startover subfunction, the distinctions between Startup and Startover and the subclasses of each.

4.1 STARTUP/STARTOVER FUNCTIONAL REQUIREMENTS

a. Initial Configuration - The Executive can configure an operational system in either a System IPL mode or a Subsystem IPL mode.

To accomplish a System IPL, a minimal configuration of elements is manually selected through switches on the Systems Console or on any Compute Element. Additional elements are configured into the operational system by the Executive, using the adapted data base to select the preferred configuration.

To accomplish a Subsystem IPL, the configuration control registers of all elements to be configured into the system must be preset to accept communications from the CE being used for IPL. Subsystem IPL implies that the elements involved have been previously configured into a subsystem.

b. Loading the NAS CFC System

1. Transferring the Startup/Startover load routines from disk to core storage (including the routines required for initial configuration.)

2. Subsequently transferring all other core resident subprograms and the data base from disk to core storage.

3. Assuring the accuracy of those transfers by initiating retries and device switching when necessary.

c. Maintenance of Internal Clocks and Calendars

The Executive will establish the initial system time in internal clocks at Startup, and maintain updated time through Startover and system recovery. It will also establish and maintain system calendars in the same manner.

d. Processing Startup/Startover Messages

e. Retrieval of Recovery Data - The Executive will:

1. Select the most recent set of recovery data.

2. Transfer the selected set of recovery data to core.

3. Verify the accuracy of the transfer of recovery data to core.

4. Put out the following information to adapted devices.

   (a) The occurrence of recovery and the mode of startover being used.

   (b) The time of the error interrupt.

   (c) The age of the recovery data being used.

f. Startup Modes - At system startup the Executive assumes an Establish Mode Startup. A Re-establish Mode startup can be selected optionally by entering an input on the 1052 printer/ keyboard or the card reader. (The Modes of Startup are described in Section 4.1.1)

g. Set Startup Mode - Process symbolic inputs (entered on the 1052 printer/keyboard or the card reader) which determine the Startup Mode, (Establish or Re-establish), and which optionally start system processing.

4.1.1 Modes of Startup

Startup is performed in one of two modes, Establish or Re-establish.

4.1.1.1 Establish Mode

Establish Mode is the normal mode of system startup. After the Executive accomplishes the functions described in subparagraphs a through d., Section 4.1, the system enters an environmental initialization interval. During this interval, inputs may be entered to initialize the environmental data base or to request printouts of stored data. The environmental interval continues until receipt of a Start Processing (GO) message.

4.1.1.2 Re-establish Mode

Re-establish Mode initializes the system and recreates the system data base with a set of recovery data selected from a previous run. The functions described in Section 4.1 are performed but no environmental initialization interval is provided. Processing continues as if a Start Processing (GO) message were entered.

4.1.2 Modes of Startover

Startover re-initializes the system for resumption of processing after an element or subprogram malfunction, or in response to an input request (STVR). Startover is performed in one of two modes, Resume or Re-establish.

4.1.2.1 Startover Initiation Methods

The following actions initiate the indicated modes of Startover:

a. Resume Mode

   1. CCC element errors that do not affect the data base.

   2. Certain types of input-requested reconfigurations of the operational system.

b. Re-establish Mode

   1. CCC element errors that affect the data base.

   2. Subprogram errors.

   3. Certain types of input-requested reconfigurations of the operational system.

   4. Input-requested startovers.

4.1.2.2 Resume Mode

The capability of starting over without resorting to stored recovery data is provided. It is used to recover from those errors which do not affect the data base. Its advantages are the minimization of data base recovery time and the retention of all inputs received before the error.

4.1.2.3 Re-establish Mode

This mode uses the recovery data to re-establish the data base to a prior condition which is adequate for the resumption of processing.

4.2 RECOVERY RECORDING DATA

The information contained on the NAS CFC System file is not sufficient to permit continuation of Central Flow Control processing after a failure which affects the contents of core storage. Startup and Startover, in the Re-establish Mode, therefore require the use of recovery data, a selected subset of dynamic data. Specific tables which constitute recovery data will be designated during the program design phase.

A set of Recovery data will be recorded periodically. The frequency of operation of this function will be parameterized. The storage medium for recovery data is disk.

Consecutive recovery data sets will be written alternately on separate disk packs. In the event of failure during the recording function, the previously recorded data set will be utilized for recovery. Each recovery data set will be time and date tagged for possible subsequent use in Re-establish Mode Startups.

4.3 INCORPORATED REFERENCES TO MONITOR HANDBOOK, SEC. 3.0

All of Monitor Handbook, Section 3.0, is incorporated by reference into this subsection of this specification.

5.0 SYSTEM ANALYSIS RECORDING

5.1 INTRODUCTION

The System Analysis Recording function transfers data from core storage to magnetic tape. The data are subsequently processed off-line for use in reconstructing and analyzing system operations. The data recorded may include Inputs, Outputs and intermediate results of internal processing. The recordings are initiated by SVC calls from the requesting programs. These requests operate in conjunction with an adapted Recording Control Table. System Analysis Recording also provides the capability to define major categories of logically associated recordings.

5.2 METHOD OF RECORDING

Recordings are normally initiated by the occurrence of significant program events. They may also be initiated periodically at predetermined intervals.

The SVC requests issued by the programs contain parameters which specify the recording. These parameters always include a recording code which identifies the data and may also include the starting address and length of the data.

The recording code associates the request with a specific entry in the Recording Control Table. It is also used in off-line processing to identify the recording.

If the starting address and length are not furnished in the SVC request, the values for these parameters are obtained from the Recording Control Table. If they are furnished, and they conflict with the values in the table, the parameters override.

The Recording Control Table also provides the capability to dynamically enable and suppress recording by the use of input messages. Requests identified by an individual recording code may be enabled or suppressed. This capability may be applied either to all requests which contain the same recording code or selectively on an individual subprogram basis. Categories of associated recording types may be defined for the purpose of assigning a specific level and scope of recording activity. The definition of these categories will be related to such factors as mode of processing, e.g., operational or test, etc., or volume of processing activity. An entire "category" of recordings may be enabled or suppressed dynamically by an input message. These categories, and the recordings which they comprise, will be defined during the design and implementation phases.

5.3 RECORDING TYPES

The System Analysis Recording function accommodates five (5) types of recording requests; table, array, dynamic, core and timing analysis (TAR).

5.3.1 Table Recordings

A Table Recording may include one or more consecutive entries. It is controlled either by optional parameters within the SVC call from the program or by an entry in the Recording Control Table.

5.3.2 Array Recordings

An Array Recording includes the entire array. It is controlled by an entry in the Recording Control Table.

5.3.3 Dynamic Recordings

5.3.3.1 Dynamic Location Recordings

The area of core storage defined by the starting address and length parameters in the SVC call is recorded.

5.3.3.2 Dynamic Queue Recordings

Subprogram queue blocks may be recorded when they are queued or unqueued, under control of an entry in the Recording Control Table.

5.3.4 Core Recordings

The area of core storage defined in the entry in the Control Table is recorded.

5.3.5 Timing Analysis Recordings (TAR's)

Timing Analysis Recordings, which are under the exclusive control of the Executive program, provide a time trace of the following significant system events:

a. Each External Interrupt

b. Each SVC Interrupt

c. Each Program Interrupt

d. Each I/O Interrupt

e. Each Start I/O

f. Each exit from the Executive to a scheduled program

g. Each occurrence of scheduled program suspension

h. Abnormal termination of a scheduled program by the Executive

i. Normal termination of a scheduled program

j. Each resumption of the scheduling function

Each TAR includes at least the following:

a. CE or IOCE identification

b. Master CE Interval Timer (12 low-order bits)

c. Event Identification, e.g., External Interrupt

d. Condition that caused the event, e.g., Timer elapsed

e. Scheduled program identification

f. Location in the operational configuration where the event occurred or to which control is transferred

Each event type requires a separate recording code. A count of lost TAR's is put out when TAR's cannot be recorded because the TAR output buffers are full.

5.4 RECORDING CONTENT

All input messages, with the exception of CTS inputs, that have passed hardware acceptance checks are recorded on the Systems Analysis Recording tape. Each message is recorded without modification. The time of receipt and the source identification are included with each message.

All output messages are recorded after being successfully transmitted. The time of completed transmission and the destination identification are included with each output message.

The recording of inputs and outputs is initiated by the Executive. The remaining data to be recorded will be specified during the design and implementation phases.

5.5 RECORDING CONVENTIONS

5.5.1 Identification of System Analysis Recordings

A data header containing at least the following, is added to each recording:

a. System time at which the System Analysis Recording function begins processing the request for a recording

b. Recording code associated with the requested recording

c. Length of the recording in bytes

d. Address of the first byte of data recorded (for the dynamic type of recording only)

e. Starting entry number and number of consecutive entries recorded, for table recordings

f. An indication of whether or not the recording is completely contained in a single physical tape record. When a recording is not completely contained in a single physical tape record, the initial, intermediate and final physical records are identified

g. An indication of data loss condition when applicable

h. An indication of Startover each time it occurs

5.5.2 Initialization of System Analysis Recording Tape Reels

Each new reel of System Analysis Recording tape is identified with a header file that contains at least the following:

a. NAS CFC System tape identity

b. Identity of the facility where the System Analysis Recording tape is produced.

c. Identity of the compool used

d. Date

e. System time

f. Reel sequence number, starting with one at startup. The reel sequence number progresses consecutively by one. Startover does not disrupt or reset the sequence number that exists before startover. Reel sequence numbering is restarted with one at midnight (Greenwich Mean Time)

g. A single tape mark (EOF)

h. Table Initialization - Depending on reduction output requirements to be specified elsewhere it may be necessary to include recordings of certain tables at the beginning of each reel. These tables will be identified during the design and implementation phases.

5.5.3 Identification of Tape Status

The System Analysis Recording function puts out messages containing the following information, on adapted output devices:

a. The time span covered by each reel of System Analysis Recording tape

b. The number of records in the data file and the reel sequence number

c. The fact that a tape reel has been closed

d. Whether End-of-File has been written on the closed tape reel

e. Any requirement for manual intervention

f. The time span during which SAR was suspended because either the output buffers were full or the output devices were temporarily unavailable

6.0 EXECUTIVE INPUT/OUTPUT MESSAGES

6.1 INTRODUCTION

This section specifies all NAS CFC Executive Input/Output messages. Subsection 6.2 incorporates by reference, those messages from the NAS EnRoute A3d2.2 program which are to be retained, as Executive messages, in the CFC program. Section 6.3 specifies new messages which are to be added for the CFC System.

6.2 INCORPORATED REFERENCES TO SELECTED NAS ENROUTE A3d2.2 DOCUMENTATION

6.2.1 NAS-MD-317

Section 6 of NAS-MD-317 is incorporated by reference into this specification, except for the following subsections:

a. 6.1.23

b. 6.1.25

c. 6.1.26

d. 6.1.27

e. 6.1.30

f. 6.1.31

g. 6.1.32

h. 6.1.33

i. 6.1.34

j. 6.2.33

k. 6.2.34

l. 6.2.36

m. 6.2.37

n. 6.2.38

o. 6.3.1.4

6.2.2 NAS Monitor Handbook

Section 7 and Appendix A of the NAS Monitor Handbook are incorporated by reference into this specification, except for the following subsections:

a. 7.5

b. 7.6.1.2, subparagraph f

c. 7.6.2.2, subparagraph o

d. 7.8

e. 7.12

f. 7.15

g. 7.16

6.2.3 NAS-MD-311

The following parts of NAS-MD-311 are incorporated by reference into this specification:

a. Section 1.0

b. Subsection 6.6

c. Subsection 6.7

d. Subsection 6.9

e. Subsection 6.20

f. Subsection 7.3

g. Subsection 7.5

h. Subsection 7.6

i. Subsection 7.7

j. Subsection 8.3

k. Subsection 8.5

l. Subsection 8.6

m. Appendix E

6.2.4 NAS-MD-314

The following parts of NAS-MD-314 are incorporated by reference into this specification:

a. Section 1.0

b. Subsection 5.2

c. Subsection 5.3.3

d. Subsection 5.3.5

e. Subsection 5.4.1

f. Subsection 5.4.3

g. Subsection 5.4.4

h. Subsection 5.4.5

i. Subsection 5.4.8

j. Subsection 5.4.9

k. Subsection 5.4.11

l. Subsection 6.3

6.2.5 NAS-MD-315

The following parts of NAS-MD-314 are incorporated by reference into this specification:

a. Section 1.0

b. Subsection 2.4.1

c. Subsection 2.4.2

d. Subsection 2.4.3

e. Subsection 2.4.4

f. Subsection 2.5.1

g. Subsection 2.5.2

h. Subsection 2.5.3

i. Subsection 6.4.1

j. Subsection 6.4.2

k. Appendix C

6.3 NEW MESSAGES

To be specified as required:

7.0 TEST AND DEBUGGING AIDS

The NAS CFC Executive Program includes the following test and debugging aids:

7.1 INPUT DATA SIMULATION

The NAS CFC operational system has the capability to respond to simulated real-time inputs from a simulation tape. The use of a simulation tape provides for controlled program testing. The simulation tape is not used during normal system operation. The simulation tape is prepared by the off-line Support System program in a format compatible with the NAS CFC operational program. Simulated inputs are read into the system at the time specified by the time parameter associated with each input on the simulation tape.

7.2 HARDCOPY PRINTOUTS OF CORE STORAGE

7.2.1 Core-to-Tape Transfer

When the execution of a subprogram or system subroutine is terminated because of a program failure (ABORT), an option permits the entire contents of core storage to be copied onto magnetic tape (Core Tape). An Off-Line Support System program prints selected areas of a core tape on the 1403 printer.

7.2.2 On-line Program Failure Printouts

The cause of most program failures (Aborts) can be determined by examining a printout of selected areas of core storage. This option makes these selected areas of core storage available in the form of an on-line 1403 printout, immediately after a program failure.

7.2.3 On-line Debugging Printouts

Any selected eight bytes of core storage may be displayed upon request from a 1052 or the 2540 card reader. In addition, selected areas of core storage may be printed on-line at previously identified instances of execution within the program.

7.3 ON-LINE MODIFICATION OF CORE STORAGE

Specified areas of core storage may be modified on-line by an input message.

7.4 MISCELLANEOUS TESTING SERVICES

In order to support system testing and debugging, the following additional capabilities are provided:

a. The execution of any operational subprogram can be initiated upon request.

b. The execution of any particular section of code can be initiated upon request.

c. The rate at which the system clocks are updated can be increased or decreased upon request.

7.5 RESOURCE MONITORING

Allocation of several major system resources is controlled by adaptation inputs. In order to provide data from which efficient adaptation values can be determined for these resources, the NAS CFC Executive can periodically record the usage of these and other internal resources. The Off-Line Support System program reduces the recorded information.

The data gathered include execution times, minimum, average and peak resource loads, and processing delays encountered due to resource depletion.

8.0 N/A

9.0 N/A

10.0 EXECUTIVE SUPERVISOR SERVICES

10.1 INTRODUCTION

This Section specifies all NAS CFC Executive Supervisor Call Services. Subsection 10.2 incorporates by reference, those supervisor calls from the NAS EnRoute A3d2.2 program which are to be retained as Executive Supervisor Call Services in the CFC program. Section 10.3 specifies new SVC's which are to be added to the CFC System.

10.2 INCORPORATED REFERENCES to MONITOR HANDBOOK, SECTIONS 2.0 & 6.0

The following parts of the Monitor Handbook are incorporated by reference into this specification:

a. Section 2.0

b. Subsection 6.3.1

c. Subsection 6.3.2

10.3 NEW SUPERVISOR CALL SERVICES

To be specified as required.

APPENDIX G - NAS A3d2.2 DOCUMENTATION

G.1 DOCUMENTATION DESCRIPTION

Since the CFC Executive functions which this document specifies are to be implemented by modifying the NAS EnRoute Monitor Program, the documentation of the EnRoute System is important to this effort. Parts of the EnRoute documentation are incorporated by reference into this specification. Section G-2 lists the specific titles and versions of any document that is either wholly or partially incorporated. Portions of the documents listed in G-2 which are not incorporated into this specification, contain other useful reference material.

Any use of shortened titles for the documents listed in G-2 are meant to refer to the specific titles and versions listed in G-2.

Section G-3 lists other EnRoute program documentation which does not contain incorporated references but does contain useful reference material.

The documents listed in Sections G-2 and G-3 have not been and will not be revised or edited to reflect the CFC functions. Any contradictions or inconsistencies between those documents and this specification will be resolved on a case by case basis.

G.2 INCORPORATED REFERENCES

The following list identifies NAS EnRoute A3d2.2 documents which contain references identified elsewhere in this specification, for incorporation into this specification. (All NAS-MD's in this list are versions current as of 1 May 1976).

a. NAS-MD-310, Introduction to Specification Series,

b. NAS-MD-311, Message Entry and Checking,

c. NAS-MD-314, Local Outputs,

d. NAS-MD-315, Remote Outputs,

e. NAS-MD-316, Adaptation,

f. NAS-MD-317, Monitor

g. NAS-MD-325, Adaptation Collection Guideline Document

h. NASP-5201-09, Monitor Handbook, Model A3d2.2, 1 December 1975.

G.3 OTHER REFERENCES

The following list identifies additional NAS EnRoute A3d2.2 documents which provide reference information about the CFC Executive program functions specified in this document:

a. NASP-5105-07 (Vol I) Program Design Specification, National Airspace System Air Traffic Control Computer Program, Volume I - Monitor Subsystem, Model A3d2.2, 1 December 1975.

b. NASP-5105-07 (Vol II), Program Design Specification, National Airspace System Air Traffic Control Computer Program, Volume II - Application Subsystem, Model A3d2.2, 1 December 1975.

c. NASP-5150 (All Volumes as of 1 May 1976), Subsystem Design Data, Monitor Model A3d2.2.

d. NASP-5155 (Volumes as of 1 May 1976) Subsystem Design Data, (selected applications subsystem volumes) Model A3d2.2

e. Compool Table Design Specifications, A3d2.2, as of 1 May 1976.

f. NASP-5231-09 or 10, Monitor Control Manual, Model A3d2.2, 1 December 1975 or 15 January 1976.

g. System Performance Analysis Report, "A General Description of the A3d2.2 (RBB) System," 1 August 1975.

h. NASP-5420-10 Data System Control and System Analysis Recording Specification, Model A3d2.2, 1 February 1976.

APPENDIX H. EXECUTIVE PROGRAM DATA BASE

A subset of the entire data base will be associated with the Executive program functions. This subset, portions of which are adaptable, is referred to in the EnRoute Monitor context as "monitor adaptation" and is described in various EnRoute documents. Incorporated references to these documents, together with deletions, additions and modifications to be specified during the design and implementation phases describe the requirements for the Executive program data base.

Some of these references describe a set of adaptable values which are defined as "system parameters". The values set forth in the listed references will be reviewed and revised during the design and implementation phases.

The following parts of the NAS EnRoute A3d2.2 documentation are incorporated by reference into this specification.

a. NAS-MD-310, Sections 1.0, 3.0 and 4.0

b. NAS-MD-316, Sections 1.0, 16.0, 17.0, 18.0 and 27.0 except:

(1) Subsection 17.6

(2) Subsection 17.7

(3) Subsection 18.4.2

(4) Subsection 18.5

(5) Subsection 18.7

(6) Subsection 18.8

c. NAS-MD-326, Sections 1.0, 5.0, 8.0 and 9.0 except:

(1) Subsection 5.6

(2) Subsection 5.7

(3) Subsection 5.8

(4) Subsection 5.9

(5) Subsection 8.17

(6) Subsection 8.18

(7) Subsection 8.19

(8) Subsection 9.1

(9) Subsection 9.2

(10) Subsection 9.3

(11) Subsection 9.6

(12) Subsection 9.7

(13) Subsection 9.8

(14) Subsection 9.11

END DOCUMENT